REMARKS

Claims 20-34 are pending. Claims 20, 26, and 30 are the independent claims. Reconsideration of the rejections is respectfully requested in view of the amendments and remarks.

The specification has been objected to, wherein the Examiner stated essentially that pages 19-20 have a small text font size.

Applicants believe the misaligned pages having a small font (pages 19-20) are the result of a scanning error at the USPTO. In order to advance prosecution of the application the above amendment to the specification re-enters the text previously found in the application. Reconsideration of the objection is respectfully requested.

Claims 20-34 stand rejected under 35 USC 102(e) as being anticipated by Messinger (US Patent No. 7,000,187). The Examiner stated essentially that Messinger teaches all of the limitations of Claims 20-34.

Referring to Claim 20:

Claim 20 claims:

A system for generating a reusable executable procedure, comprising:

a client device comprising an application for monitoring and recording a procedure that is performed using said client device and generating an execution trace representing an instance of said procedure;

a procedure trace repository for storing stored execution traces associated with instances of said procedure; and

a server for processing said execution trace and said stored execution traces to generate said reusable executable procedure, wherein said procedure can be automatically performed on the client by invoking the reusable executable procedure.

Messinger teaches a method for displaying a sequence of instructions associated with a task in a graphical overlay, for example, for software training (see Abstract). Messinger does not teach "a server for processing said execution trace and said stored execution traces to generate said reusable executable procedure." Messinger merely teaches how an "on-line coach" operates based on user (trainee) input (see col. 2, lines 4-8). An "on-line coach" is not analogous to "a server for processing said execution trace and said stored execution traces to generate said reusable executable procedure," as claimed in Claim 20. For example, nowhere does Messinger teach or suggest how an "on-line coach" could be generated, much less "processing said execution trace and said stored execution traces to generate said reusable executable procedure," as claimed in Claim 20. Therefore, Messinger fails to teach all the limitations of Claim 20.

Regarding the Examiner's comments at page 4 (second paragraph) that "a network administrator controls/administrates the whole process"; respectfully, it is beneficial here to return to the premise of Messinger, that of an overlay positioned on top of a GUI window, wherein the overlay displays instructions upon used selection of a task (see Abstract). Messinger at col. 12, lines 28-52 relates to a level of coaching - that is, tasks displayed correspond to a level of user-privilege; for example, a network administrator would have more tasks displayed than a lower-level user. Nowhere does Messinger teach that a network administrator "controls/administrates the whole process" as suggested in the rejection.

Messinger fails to teach how an "on-line coach" could be generated. Thus, Messinger does not teach all of the limitations of Claim 20.

Referring now to Claim 26:

Claim 26 claims:

A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for generating a reusable executable procedure, the method steps comprising:

obtaining a plurality of execution traces, wherein each execution trace represents an execution instance of a procedure; and

processing said execution traces to create a reusable executable procedure associated with said procedure, wherein said procedure can be automatically performed by invoking the reusable executable procedure.

Messinger teaches a method for displaying a sequence of instructions associated with a task in a graphical overlay, for example, for software training (see Abstract). Messinger does not teach "obtaining a plurality of execution traces, wherein each execution trace represents an execution instance of a procedure; and processing said execution traces to create a reusable executable procedure associated with said procedure" as claimed in Claim 26. Although Messinger arguably teaches a process of recording a new task sequence (in block 385 of FIG. 8), even assuming that the process of recording a new task sequence is an execution trace that represents an execution instance of a procedure, Messinger fails to teach "obtaining a plurality of execution traces, wherein each execution trace represents an execution instance of a procedure; and processing said execution traces to create a reusable executable procedure associated with

said procedure" - for example, Messinger merely adds a new task to a task list. Each task of Messinger is individually maintained and there is no processing to create a reusable executable procedure, essentially as claimed. Therefore, Messinger fails to teach all the limitations of Claim 26.

Referring to Claim 30:

Claim 30 claims:

A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for executing a reusable procedure, the method steps comprising:

launching a reusable executable procedure;

automatically executing procedure steps associated with said reusable executable procedure; and

relinquishing control of execution of said reusable procedure to a user, when a next step of said reusable executable procedure cannot be successfully executed.

Messinger teaches a method for displaying a sequence of instructions associated with a task in a graphical overlay, for example, for software training (see Abstract). Messinger does not teach "automatically executing procedure steps associated with said reusable executable procedure; and relinquishing control of execution of said reusable procedure to a user, when a next step of said reusable executable procedure cannot be successfully executed" as claimed in Claim 30. Messinger's overlay merely suggests instructions upon a user selection of a task (see col. 6, lines 4-12). For example, Messinger simply highlights a location of user action (see box 235, FIG. 6B). Nowhere does Messinger teach that the overlay automatically executing

procedure steps associated with said reusable executable procedure for a user, much less relinquishing control of execution of said reusable procedure to a user, essentially as claimed in Claim 30. Therefore, Messinger fails to teach all the limitations of Claim 30.

For the foregoing reasons Claims 20, 26, and 30 are believed to be allowable over Messinger. Claims 21-25 depend from Claim 20. Claims 27-29 depend from Claim 26. Claims 31-34 depend from Claim 30. The dependent claims are believed to be allowable for at least the reasons given for the respective independent claims. Reconsideration of the rejection is respectfully requested.

Claims 20 and 26 stand rejected under 35 USC 102(e) as being anticipated by Bala (US Patent Application No. 2004/0130572). The Examiner stated essentially that Bala teaches all of the limitations of Claims 20-34.

Referring to Claim 20:

Claim 20 claims:

A system for generating a reusable executable procedure, comprising:

a client device comprising an application for monitoring and recording a procedure that is performed using said client device and generating an execution trace representing an instance of said procedure;

a procedure trace repository for storing stored execution traces associated with instances of said procedure; and

a server for processing said execution trace and said stored execution traces to generate said reusable executable procedure, wherein said procedure can be automatically performed on the client by invoking the reusable executable procedure.

Bala teaches methods for authoring and executing wizards, wherein wizards are updated through a feedback system (see Abstract). Bala fails teach "a server for processing said execution trace and said stored execution traces to generate said reusable executable procedure, wherein said procedure can be automatically performed on the client by invoking the reusable executable procedure" as claimed in Claim 20. Bala merely teaches how a user can create script for a task (see paragraphs [0074-000079]). Bala is totally devoid of description related multiple scripts, much less "processing said execution trace and said stored execution traces to generate said reusable executable procedure," as claimed in Claim 20. Therefore, Bala fails to teach all the limitations of Claim 20.

Referring now to Claim 26:

Claim 26 claims:

A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for generating a reusable executable procedure, the method steps comprising:

obtaining a plurality of execution traces, wherein each execution trace represents an execution instance of a procedure; and

processing said execution traces to create a reusable executable procedure associated with said procedure, wherein said procedure can be automatically performed by invoking the reusable executable procedure.

Bala teaches methods for authoring and executing wizards, wherein wizards are updated through a feedback system (see Abstract). Bala does not teach the process of "obtaining a plurality of execution traces, wherein each execution trace represents an execution instance of a procedure; and processing said execution traces to create a reusable executable procedure associated with said procedure" as claimed in Claim 26. Bala merely teaches how a user can create script for a task (see paragraphs [0074-000079]). Bala is totally devoid of description related multiple scripts, much less "obtaining a plurality of execution traces, wherein each execution trace represents an execution instance of a procedure; and processing said execution traces to create a reusable executable procedure associated with said procedure," as claimed in Claim 26. Therefore, Bala fails to teach all the limitations of Claim 26.

The Examiner's reconsideration of the rejection is respectfully requested.

Claim 30 stands rejected under 35 USC 102(b) as being anticipated by Mayuzumi (US Patent No. 6,134,644). The Examiner stated essentially that Mayuzumi teach all of the limitations of Claim 30.

Claim 30 claims:

A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for executing a reusable procedure, the method steps comprising:

launching a reusable executable procedure;

automatically executing procedure steps associated with said reusable executable procedure; and

relinquishing control of execution of said reusable procedure to a user, when a next step of said reusable executable procedure cannot be successfully executed.

Respectfully, the anticipation rejection of Claim 30 is based on a mischaracterization of the teachings of Mayuzumi as applied to claim 30. There is no reasonable basis, whatsoever, for construing Mayuzumi (FIG. 16 and supporting explanation in Col 20, lines 15-29 and Col. 21, lines 1-17) as teaching the claimed process of "launching a reusable executable procedure; automatically executing procedure steps associated with said reusable executable procedure; and relinquishing control of execution of said reusable procedure to a user, when a next step of said reusable executable procedure cannot be successfully executed."

Mayuzumi does not specifically teach a "reusable executable procedure" within the context of the claimed inventions. Mayuzumi teaches a method for dynamically generating a sequence of data parts based on the type of help information needed at a given time (i.e., depending on a given event or combination of events) (see, e.g., Col. 19, lines 35 – Col. 20, line 9). In this regard, the sequence of data parts is not maintained as "reusable executable procedures" per se, as the sequence of data parts is dynamically generated.

Moreover, Mayuzumi teaches in FIG. 16 a method in which help screens are presented to a user illustrating a process flow for sequential work that can be performed to recover from an error (see, Col. 20, lines 52-67, FIG. 15). In this regard, the process flows are not automatically executed as part of a reusable executable procedure, essentially as claimed in Claim 30, but are merely dynamically generated based on one or more events, and then displayed to provide user guidance in manually performing a given task as suggested in the displayed process flow. Consider that the method of Mayuzumi checks for user action at each of blocks S13, S14, S18, and S19 - this is illustrative of the fact that Mayuzumi does not take automatic action with respect to executing procedure steps, much less relinquishing control of execution of said reusable procedure to a user. Therefore, Mayuzumi fails to teach all the limitations of Claim 30.

Reconsideration of the rejection is respectfully requested.

For the forgoing reasons, the present application, including Claims 20-34, is believed to be in condition for allowance. The Examiner's early and favorable action is respectfully urged.

Respectfully submitted,

Dated: April 22, 2008

/Nathaniel T. Wallace/
Nathaniel T. Wallace
Reg. No. 48,909
Attorney for Applicants

Mailing Address:
F. CHAU & ASSOCIATES, LLC
130 Woodbury Road
Woodbury, New York 11797
TEL: (516) 692-8888
FAX: (516) 692-8889